WHAT IS CLAIMED IS:

1.   A microprocessor having a unique secret key and a
unique public key corresponding to the unique secret key
that cannot be read out to external, comprising:

a reading unit configured to read out a plurality of
programs encrypted by using different execution code
encryption keys from an external memory;

a decryption unit configured to decrypt the plurality
of programs read out by the reading unit by using
respective decryption keys;

an execution unit configured to execute the plurality
of programs decrypted by the decryption unit;

a context information saving unit configured to save a
context information for one program whose execution is to
be interrupted, into the external memory or a context
information memory provided inside the microprocesor, the
context information containing information indicating an
execution state of the one program and the execution code
encryption key of the one program; and

a restart unit configured to restart an execution of
the one program by reading out the context information from
the external memory or the context information memory, and
recovering the execution state of the one program from the
context information.

2.   The microprocessor of claim 1, wherein the context
information saving unit is configured to encrypt the
context information by using the public key, and to save an
encrypted context information into the external memory; and

the restart unit is configured to restart the
execution of the one program by reading out the encrypted
context information from the external memory, decrypting
the encrypted context information by using the secret key,
and recovering the execution state of the one program from

a decrypted context information.

3. The microprocessor of claim 2, wherein the restart
unit restarts the execution of the one program only when a
decrypted execution code encryption key contained in the
decrypted context information coincides with the execution
code encryption key of the one program.

4. The microprocessor of claim 2, wherein the restart
unit uses a decrypted execution code encryption key
contained in the decrypted context information as a
decryption key for decrypting the one program.

5. The microprocessor of claim 1, wherein the context
information saving unit is configured to save the context
information in a plaintext form into the context
information memory which is not readable by another program
which is executed after the one program is interrupted; and
the restart unit is configured to restart an execution
of the one program by reading out the context information
from the context information memory, and recovering the
execution state of the one program from the context
information.

6. The microprocessor of claim 5, wherein the restart
unit restarts the execution of the one program in response
to an execution of a prescribed instruction by the another
program.

7. The microprocessor of claim 6, wherein the context
information saving unit saves the context information into
the context information memory at a time of interrupting
the execution of the one program, and encrypts the context
information in the context information memory by using the
public key and stores the encrypted context information

into the external memory in response to an execution of
another prescribed instruction by the another program.

8.   The microprocessor of claim 5, wherein the context
5 information saving unit saves the context information into
the context information memory at a time of interrupting
the execution of the one program, and encrypts the context
information in the context information memory by using the
public key and stores the encrypted context information
10 into an address on the external memory that is specified by
the another program.

9.   The microprocessor of claim 1, wherein the context
information saving unit is configured to generate a random
15 number as a temporary key, to encrypt the context
information, and to save an encrypted context information
into the external memory, the encrypted context information
containing a first value obtained by encrypting information
indicating the execution state of the one program by using
20 the temporary key and a second value obtained by encrypting
the temporary key by using the public key; and
        the restart unit is configured to restart the
execution of the one program by reading out the encrypted
context information from the external memory, decrypting
25 the temporary key from the second value contained in the
encrypted context information by using the secret key,
decrypting the information indicating the execution state
from the first value contained in the encrypted context
information by using a decrypted temporary key, and
30 recovering the execution state of the one program from a
decrypted context information.

10.   The microprocessor of claim 9, wherein the context
information saving unit saves the encrypted context
35 information that also contains a third value obtained by

encrypting the temporary key by using the execution code
encryption key of the one program.

11.   The microprocessor of claim 10, wherein the restart
unit decrypts a first temporary key from the second value
contained in the encrypted context information by using the
secret key and decrypts the information indicating the
execution state from the first value contained in the
encrypted context information by using the first decrypted
temporary key, while decrypting a second temporary key from
the third value contained in the encrypted context
information by using the execution code encryption key of
the one program, and restarts the execution of the one
program only when the first decrypted temporary key
coincides with the second decrypted temporary key.

12.   The microprocessor of claim 1, further comprising:
        an execution state memory unit for storing an
execution state of a currently executed program; and
        an execution state initialization unit configured to
initialize a content of the execution state memory unit to
a prescribed value or encrypts the content of the execution
state memory unit, before an execution of another program
starts after the one program is interrupted.

13.   The microprocessor of claim 1, further comprising:
        a key reading unit configured to read out the
execution code encryption key of each program that is
encrypted by using the public key in advance, from the
external memory; and
        a key decryption unit configured to decrypt the
execution code encryption key read out by the key reading
unit, by using the secret key;
        wherein the decryption unit decrypts each program by
using the execution code encryption key as a decryption

key.

14. The microprocessor of claim 1, further comprising:
        an execution state memory unit for storing an
5 execution state of a currently executed program and an
    encryption attributes for data to be processed by the
    currently executed program; and
        a data encryption unit configured to encrypt the data
    to be processed by the currently executed program according
10 to the encryption attributes stored in the execution state
    memory unit.

15. The microprocessor of claim 1, further comprising:
        an execution state memory unit for storing an
15 execution state of a currently executed program, encryption
    attributes for data to be processed by the currently
    executed program, and an encryption attribute specifying
    information for specifying the encryption attributes;
        a related information writing unit configured to write
20 a related information related to the encryption attribute
    specifying information and containing a signature obtained
    by using the secret key, into the external memory;
        a related information reading unit configured to read
    out the related information from the external memory
25 according to an address of a data to be referred by the
    currently executed program;
        a data referring permission unit configured to verify
    the signature contained in the related information by using
    the public key, and to permit a data referring by the
30 currently executed program by determining an encryption key
    and an algorithm to be used for the data referring
    according to the related information and the encryption
    attribute specifying information, only when the signature
    contained in the related information coincides with an
35 original signature of the microprocessor; and

a data encryption unit configured to encrypt the data
to be referred by the currently executed program according
to the encryption attributes stored in the execution state
memory unit.

16. The microprocessor of claim 1, further comprising:

a cache memory for caching plaintext instructions and
plaintext data for the plurality of programs in units of
cache lines, the cache memory having an attribute area for
each cache line indicating a decryption key identifier for
uniquely identifying a decryption key used in decrypting
each program whose instructions are cached in each cache
line or each program whose execution has caused caching of
the plaintext data in each cache line;

a cache access control unit configured to permit a
data referring caused by an execution of one cached program
stored in one cache line with respect to one cached data in
another cache line, only when the decryption key identifier
indicated by the encryption attribute for the one cache
line coincides with the decryption key identifier indicated
by the encryption attribute for the another cache line.

17. The microprocessor of claim 16, wherein when the data
referring is not permitted, new data are cached into the
another cache line from the external memory.

18. The microprocessor of claim 16, wherein when the data
referring is not permitted, an execution of the one cached
program is interrupted by a protection exception.

19. The microprocessor of claim 1, wherein the execution
unit also executes plaintext programs, and has a debugging
function for causing an exception when an execution of a
program at a specific address or address region or a data
referring to a data at the specific address or address

region occurs during an execution of a plaintext program, the debugging function being invalidated during an execution of an encrypted program.

5   20.   The microprocessor of claim 1, wherein constituent elements of the microprocessor are contained in a single chip or a single package.

10

15

20

25

30

35